

and be2, then ae1 is inside P(B) and can be selected as the starting edge of the polygon intersection. If ae2 and ae1 both are not on the right side of be1 and be2, there is no intersection at N (as shown in Figure 7B). To determine whether ae2 is on the right side of be1 and be2, the process 500 shown in Figure 8 can be used. Using the process 500 in Figure 8, the shape point bp1 is selected from be1, the shape point bp2 is selected from be2 and the shape point ap2 is selected from ae2. These shape points are selected because they are the closest shape points on their respective edges to node N (i.e., point p) although other shape points on these edges may be selected as well. The process 500 in Figure 8 is used to test if point ap2 is on the right side of the line segment bp1_p and p_bp2. According to the process 500, to test the position of a point, P3, relative to a line segment P1P2, calculate $c = (x1 - x3)*(y2 - y3) - (x2 - x3) * (y1 - y3)$. If $c = 0$, P3 is on the line segment P1P2. If $c > 0$, P3 is on right side of the line segment P1P2. If $c < 0$, P3 is on left side of the line segment P1P2. If ap2 is on the right side of the line segment bp1_p and p_bp2, ae2 is to the right of be1 and be2 (as shown in Figure 7A). If not, ae2 is not on the right of be1 and be2 (as shown in Figure 7B).

At page 8, replace the paragraph starting at line 8 and continuing through line 12 with the following corrected paragraph. —

If there are two or more other links that connect to the clockwise end of the LINK(START), then the link that forms the smallest angle in the counterclockwise direction from LINK(START) is the next link of the intersection polygon P(I). Data indicating this next link are stored in the list formed to represent the intersection polygon P(I) (Steps 272, 278 and 276).

IN THE CLAIMS (clean versions):

Please amend Claims 1, 10, 14 and 18, as indicated.

1. (Amended) A method for determining a polygonal intersection of a first polygon and a second polygon comprising:
- at an intersection of a boundary of the first polygon with a boundary of the second polygon, determining a first portion of a boundary of the polygonal intersection as

5 comprised of a portion of the boundary of the first polygon that is located inside the
6 second polygon; and
7 determining each subsequent portion of the boundary of the polygonal
8 intersection by selecting a portion of the boundary of either the first polygon or the
9 second polygon that connects to a leading end of a portion of the boundary of the
10 polygonal intersection that had previously been determined and that forms a minimum
11 rotation angle in a selected direction therewith.

12

1 10. (Amended) The method of Claim 1 wherein the portion of the boundary
2 of the first polygon that is located inside the second polygon is determined by
3 determining on which side of the boundary of the second polygon the portion of the
4 boundary of the first polygon is located.

5

1 14. (Amended) A program for determining a polygonal intersection of a
2 first polygon and a second polygon, wherein said program is stored on a computer-
3 readable medium, said program comprising:

4 program code that determines a first portion of a boundary of the polygonal
5 intersection as being comprised of a portion of a boundary of the first polygon that is
6 located inside the second polygon at an intersection of the boundary of the first polygon
7 with a boundary of the second polygon; and

8 program code that determines each subsequent portion of the boundary of the
9 polygonal intersection by selecting a portion of the boundary of either the first polygon or
10 the second polygon that connects to a leading end of a portion of the boundary of the
11 polygonal intersection that had previously been determined and that forms a minimum
12 rotation angle in a selected direction therewith.

13

1 18. (Amended) A method for determining a polygonal intersection of a first
2 polygon and a second polygon represented by data contained in a geographic database,
3 wherein a boundary of the first polygon is represented by a first list of links
4 connected at endpoints thereof and the second polygon is represented by a second list of
5 links connected at endpoints thereof,

6 wherein an endpoint of a link is represented by either a node or a shape point;
7 wherein each location at which the boundary of the first polygon intersects with
8 the boundary of the second polygon is represented by a node;
9 wherein the links contained in the first list of links are in an order corresponding
10 to a consistent direction of traversal of the corresponding links representing the boundary
11 of the first polygon;
12 wherein the links contained in the second list of links are in the order
13 corresponding to the consistent direction of traversal of the corresponding links
14 representing the boundary of the second polygon;
15 the method comprising the steps of:
16 determining a first minimum bounding rectangle that encompasses the first
17 polygon;
18 determining a second minimum bounding rectangle that encompasses the second
19 polygon;
20 determining that the first minimum bounding rectangle and the second minimum
21 bounding rectangle intersect;
22 identifying all the links located entirely in a first polygonal area formed by an
23 intersection of the first minimum bounding rectangle and the second minimum bounding
24 rectangle that have at least one node at an endpoint thereof;
25 associating in a node-link map each node connected to each of the identified links
26 with each of the links connected thereto;
27 identifying a node from the node-link map that has at least three links connected
28 thereto;
29 from the order of two of said at least three links that belong to one of the
30 polygons, determining which one of said at least three links that belong to the other of
31 polygons is located inside the one of said polygons;
32 using the link that is located inside the one of said polygons as a starting link for
33 the polygonal intersection of the first polygon and the second polygon; and
34 determining each other link of the polygonal intersection by selecting from the
35 links that connect to a currently known link at the end thereof according to the consistent